RFC 757

A Suggested Solution to the Naming, Addressing, and Delivery Problem for ARPAnet Message Systems

Debra P. Deutsch

10 September 1979

Bolt Beranek and Newman

50 Moulton Street

Cambridge, Massachusetts 02138

(617) 491-1850

Table of Contents

1. Introduction 2

2. Names and Addresses 3

2.1 A distributed database approach 4
2.2 Delivery 5
2.2.1 Additional delivery options 7
2.2.2 Failures 8

3. Relationship to TIP Login database 9

4. Relationship to RFC 753 10

4.1 Compatibility 10
4.1.1 Outgoing message 10
4.1.1.1 RFC 753 10
4.1.1.2 This proposal 11
4.1.2 Messages in transit 12
4.1.3 Incoming mail 12

5. Implications of an internetwork message environment 13

5.1 Birthplaces 13
5.1.1 ID resolution 13
5.1.2 Hosts in an internet environment 13
5.1.3 Orphans 14

6. Conclusions 15

List of Figures

Figure 4-1: The Internet Environment 10

Preface

Unlike many RFCs, this is not a specification of a soon-to-be-implemented protocol. Instead this is a true request for comments on the concepts and suggestions found within this document, written with the hope that its content, and any discussion which it spurs, will contribute towards the design of the next generation of computer-based message creation and delivery systems.

A number of people have made contributions to the form and content of this document. In particular, I would like to thank Jerry Burchfiel for his general and technical advice and encouragement, Bob Thomas for his wisdom about the TIP Login database and design of a netmail database, Ted Myer for playing devil's advocate, and Charlotte Mooers for her excellent editorial assistance.

Debbie Deutsch

1. Introduction

The current ARPAnet message handling scheme has evolved from rather informal, decentralized beginnings. Early developers took advantage of pre-existing tools -- TECO, FTP -- in order to implement their first systems. Later, protocols were developed to codify the conventions already in use. While these conventions have been able to support an amazing variety and amount of service, they have a number of shortcomings.

One difficulty is the naming/addressing problem, which deals with the need both to identify the recipient and to indicate correctly a delivery point for the message. The current paradigm is deficient in that it lacks a sharp distinction between the recipient's name and the recipient's address, which is the delivery point on the net.

The naming/addressing scheme does not allow users to address their messages using human names, but instead forces them to employ designations better designed for machine parsing than human identification.

Another source of limitations lies in the delivery system, which is simply an extension of the File Transfer Protocol. The delivery system is fairly limited in its operation, handling only simple transactions involving the transfer of a single message to a single user on the destination host. The ability to bundle messages and the ability to fan-out messages at the foreign host would improve the efficiency and usefulness of the system.

An additional drawback to the delivery system is caused, to some extent, by the addressing scheme. A change in address, or incorrect address usually causes the delivery system to handle the message incorrectly. While some hosts support some variety of a mail forwarding database (MFDB), this solution is at best inadequate and spotty for providing reliable service to the network as a whole. Because the same username may belong to different people at different hosts, ambiguities which may crop up when messages are incorrectly addressed keep even the best MFDBs from always being able to do their job.

This proposal envisions a system in which the identity and address of the recipient are treated as two separate items. A network database supports a directory service which supplies correct address information for all recipients. Additionally, the scheme allows mail delivery to be restricted to authorized users of the network, should that be a desirable feature.

2. Names and Addresses

Today's ARPAnet naming and addressing scheme (as specified in RFC 733[3]) does not discriminate between the identity of a user and his address[1]. Both are expressed the same way: USERNAME@HOST. While this should always result in a unique handle for that user, it has proved to be inadequate in practice. Users who change the location of their mailboxes, because of either a change in affiliation or a simple shift in host usage, also get their names changed. If their old host employs an MFDB the problem is not too bad. Mail is simply forwarded on to the new address, slightly delayed. Other less fortunate users who cannot rely upon an MFDB must notify all their correspondents of the change in address/name. Any mail addressed to the old address becomes undeliverable. (An excellent discussion of the differences between naming, addressing, and routing is found in a paper by John Shoch [1].)

The desire to use "real" names in the address fields of messages is also thwarted by the current system. An elaborate system for using human-compatible vs. machine-interpretable information has been evolved for use in message headers[2]. The most recent developments indicate that many users would feel happiest if the real human name could appear; machine-interpretable information should not intrude too heavily into the writer's work- and thought-space.

The solution proposed here calls for a total break between the way a recipient is named or identified and the way in which his location is specified. Since the ARPAnet is a regulated environment, a unique (and not necessarily human-readable) ID could be assigned to each authorized recipient of network mail. This ID would stay with the user throughout his lifetime on the network, through changes in address and affiliation.

A network database (which could be derived from the same database that has been proposed to support TIP login) would associate each ID with one or more addresses indicating where the mail for that ID may be delivered. If more than one address were

---

1 See, for example, RFC 733's discussion of the semantics of address fields, in which it is specified that the To: field "contains the identity of the primary recipients of the message".

2 See the "Syntax of General Addressee Items" section of RFC 733.

associated with an ID, that would indicate an ordered preference in delivery points. The delivery system would attempt delivery to the first addressee, and, if that failed, try the second, and so on [3]. Most IDs would probably have only one address. Also associated with each ID would be some information about the ID's owner: name, postal address, affiliation, phone number, etc.

Rather than being forced to type some awkward character string in order to name his correspondent, the writer would have to supply only enough information to allow some process to determine the unique identity of the recipient. This information might be the recipient's name or anything else found in the database.

The access to this data would also free the writer from any need to know the location of the recipient. Once the unique ID were known, the correct location for delivery would be only a look-up away.

2.1 A distributed database approach

It is clear that if the network database had only one instantiation there would be a tremendous contention problem. All message traffic would be forced to query that one database. This is extremely undesirable, both in terms of reliability and speed. It is also clear that requiring each host to maintain a complete local copy of the entire network database is an undesirable and unnecessary burden on the hosts.

A better approach would be to build some sophistication into the local delivery system, and use local mini-databases which are based upon the contents of a distributed network database. (It may be redundant and/or partitioned, etc., but is probably not resident on the local host.) When a local host queries the network database about an ID (or, in a more costly operation, asked to supply an ID given enough identification such as name, etc.) the database may be asked to return all its information on that ID. At this point the local host can enter all or some of that information into a locally maintained database of its own. It will always refer to that database first when looking for a name or address, only calling the network database if it cannot find a local entry. Depending upon the desired level of sophistication of the local message handling programs, additional information may be added to that database, including, for

---

[3] Multiple addresses might also be used to indicate that multiple deliveries are desired.

example, nicknames.

The database might be shared by a cluster of hosts (such as exist at BBN or ISI), or it might be used by only one. Hosts which originate small amounts of message traffic might rely upon the network database entirely.

The structure and maintenance of the local databases is left solely to the local hosts. They may or may not store addresses. It may be desirable either to garbage collect them, or to let them grow. The local databases might be linked to smaller, more specialized databases which are owned by individual users or groups. These individual databases would be the equivalent of address books in which users might note special things about individuals: interests, last time seen, names of associates, etc. The existence and scope of these databases are not mandated by this scheme, but it does allow for them.

The same individual databases may be used by message creation programs in order to determine the recipient's ID from user-supplied input. For example, a user may address a message to someone named Nick. The message creation program may associate "Nick" with an ID, and hand that ID off to the delivery system, totally removing the matter of address or formal ID from the user's world.

2.2 Delivery

The delivery operation consists of three parts:

1. Determining the address to which the message must be sent,

2. Sending the message,

3. Processing by foreign host.

The first step usually means looking up, in either a local or the network database, the correct address(es) for message delivery, given the recipient's ID. Should the ID not be known at the time the message is submitted for delivery, any operation necessary to determine that ID (such as a call to either the local or network database) is also performed as part of this step.

The second step is not too different from what happens today. The local host establishes a connection to the foreign host. It is then able to send one or messages to one or more people. The options are:

- Bulk mail. Several recipients all get the same message.

- Bundled mail. Several messages get sent to the same recipient.

- A combination of the above

- One recipient gets one message.

The foreign host should be able to accept mail for each ID.

The rejection of mail for a given ID by the foreign host would usually indicate an inconsistency between the sender's local database and the network database. In this case, the local host updates its local database from the network database, and attempts delivery at the "new" host. (This is mail forwarding.) If a host taken from the network database is found to be incorrect, there is a problem in the network database, and appropriate authorities are notified. Thus, address changes propagate out from the network database only as the out-of-date information is referenced. This reduces the magnitude of the local database update problem.

Once the foreign host recognizes the ID(s), the message(s) may be transmitted to the foreign host. Upon successful transmission, the job of the local host is done.

The third step requires the foreign host to process the message(s). This is analogous to what may occur in a mail room. A foreign host may have to sort the bundled or bulk mail it receives. In addition, the foreign host might perform internal or external fan-out functions or other special functions, at the option of the ID owner.

The implemention and design of possible functions which may be performed in the mail rooms are neither mandated nor restricted by this delivery scheme. Since they are too numerous to allow even a small portion of them to be described here, only a few examples will be mentioned.

Fan-out functions might include placing messages in multiple files, sending copies to one or more other users, or rebroadcasting the messages onto the network. (In that last case, the foreign host might evaluate an ID list, in much the same way that the ITS mail repeater broadcasts messages addressed to certain mailboxes.) Special functions might include automatic hard-copy creation or reply generation, processing by various daemons, or any other service found desirable by the host's user population and administration. The implementation of fan-out functions is up to the local host, as are any additional functions which the user population might wish of its local "mail room". Whatever services are available, the mail room will distribute the mail to the correct location for each ID.

2.2.1 Additional delivery options

It may be desirable to allow mail rooms to accept a username in place of an ID. Use of a username is a less reliable method of addressing than use of an ID.

- A username may not be sufficiently unambiguous for getting an ID and host from the network database.

- Since a recipient's username may change from time to time, there is a chance that the username supplied by the sender will be incorrect [4], or that the host may not recognize it.

  Because a recipient's ID does not change with time, errors such as those caused by username changes cannot occur if IDs are used. Similarities or ambiguities can be discovered before delivery occurs, and the sender can be prompted for additional identifying information about his intended recipient.

- In an even worse case, a correct username can still result in an incorrect delivery when it is paired with an incorrect host or acted upon by a mail forwarding database [5].

  Because unique IDs are unambiguous, the possibility of such a situation is eliminated by the use of unique IDs.

---

[4] A particularly insidious source of addressing errors stems from the inconsistent use of (human) names and initials to generate usernames. The sender can easily guess his recipient's username incorrectly by using, or failing to use a combination of initials and last name. (For example, a user wishing to address Jim Miller at BBNA and using the address "Miller@BBNA" will have his message successfully delivered to Duncan Miller at the same site.)

[5] The author has observed a mail forwarding database redirect messages correctly addressed to one JWalker to different JWalker at another host.

2.2.2 Failures

The case in which the network database is found to be incorrect has already been discussed. It may make sense to mark the entry as "possibly in error" and to notify both the network database and the ID owner[6] when such a situation occurs. In this case mail delivery to the ID's owner will not occur, but this is not too bad, considering that that is what happens today when a host does not recognize a username.

One additional failure mode, the loss of the network database from the net, must be considered, even though a well-designed distributed network database should be robust enough to almost rule out this possibility.

If such a failure should occur, the local databases should be able to handle most of the traffic. What would be lost is the ability to add new IDs to the network database, the ability to change hosts for an ID, the ability to update local databases, and the ability to query the network database. In essence, there would be a regression to the state we are in today.

A well-administered network database should be backed up frequently. Should a catastrophic series of hardware failures remove one or more of the network database's hosts from the net, the database could be moved elsewhere. Such a change would entail notification of all hosts on which mail originates. Software which queries the database should be designed to be able to easily handle such a move.

---

[6] Such notification would presumably be by hardcopy mail or telephone.

3. Relationship to TIP Login database

A number of references to the TIP Login problem and a database which has been proposed as part of its solution have been made in this note. A series of working papers [5] written by Bob Thomas, Paul Santos, and Jack Haverty describe an approach to TIP Login. In brief, the method is to build and maintain a distributed TIP Login database, containing information necessary to allow a new entity called a "login-host" to decide whether or not to grant a user access to a given TIP, and whether or not to allow the user to make various modifications to the database itself.

The TIP login database is derived from a "network user data base", which contains information above and beyond that necessary to support TIP login. This comprehensive database is designed to support applications other than TIP Login, either directly or by means of databases derived from it.

Contained in the TIP Login database are each user's login string, a list of TIPs the user is authorized to access, the user's unique ID, his password, and any other "permissions" (in addition to which TIPs may be accessed). These permissions may indicate that the user may create, delete, or modify entries in the database, to assume other user's roles, and to what extent he may do so. The notion of permissions as developed by Steve Warshall is discussed in an NSW memo [2].

It seems entirely reasonable to derive a netmail database from the same comprehensive database that is designed to support TIP Login. The concept of a unique ID is supported by that database. Much of the required information for a netmail database is already included, and the maintenance tools necessary to modify it seem well-suited for the purpose. The concept of permissions extends well to the needs of netmail. Permissions specific to network mail might include, for example, the ability to modify the delivery host list associated with a given user.

The mechanisms necessary for the maintenance of the comprehensive network database and its derived databases give us a netmail database very inexpensively. This proposal takes advantage of that situation.

4. Relationship to RFC 753

RFC 753 [4] describes an internetwork message delivery system. Very briefly, the approach is to locate one or more "message processing modules" (or MPMs) on each network. These MPMs pass messages across network boundaries, and are also capable of making deliveries to users on the local network. The document also details a proposed message format, along the envelope and letter paradigm. An external "envelope", read by the delivery system, allows the (unread) message to be correctly routed and delivered to the proper recipient. Groups of messages passed between a pair of MPMs are sent together in a "mail bag".

This proposal differs from RFC 753 in that it is primarily intended to operate within a network or a concatenation of networks using a common host-host protocol, e.g. TCP. Where RFC 753 addresses the problems of internetwork communication (differing message formats, complex routing, and correct identification of the proper recipient), this note concentrates primarily on what can be done within a single protocol. The two are not incompatible. While a general internetwork protocol must provide general methods which can be compatible with different host-host protocols in different networks, a proposal such as this one can capitalize on the capabilities, resources, and policies of a given catenet (catenated network) such as the ARPAnet/PRnet/SATNET etc.

4.1 Compatibility

The delivery system described in RFC 753 is compatible with the system outlined here. Let's examine this for each of the three basic delivery options performed by the MPM. (In the discussion that follows, "local networks" means a concatenation of networks using a common host-host protocol, e.g. TCP. "Foreign network" means some network which uses a different host-host protocol, e.g. X.25. (See Figure 4-1.)

4.1.1 Outgoing message

4.1.1.1 RFC 753

The sender's process hands a message to the local network MPM. The message may be destined to an address on the local network or on a foreign network. In the former case, the MPM performs the local delivery function (see "Incoming message"). In the latter case, the MPM passes the message along to another MPM which is "closer" to the end user.

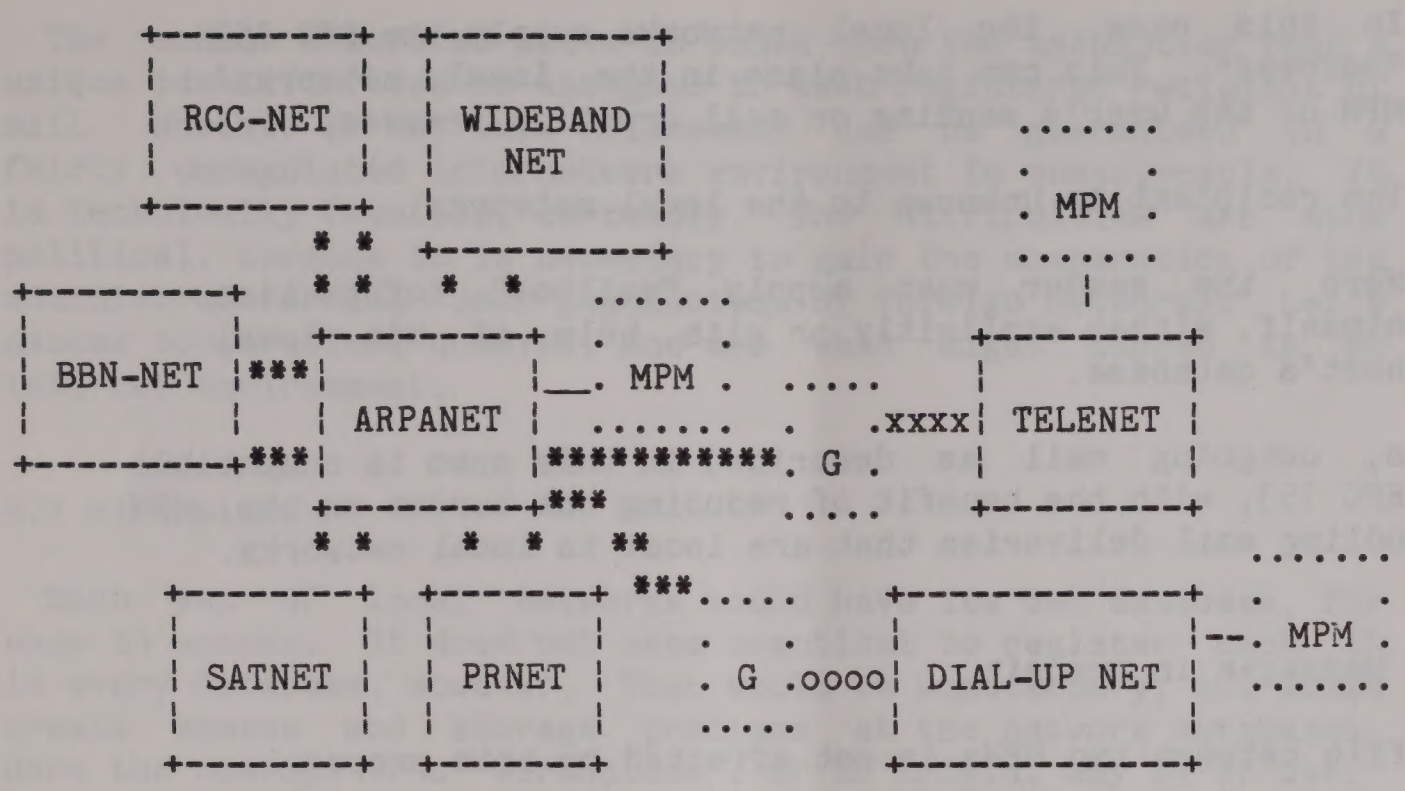

Figure 4-1: The Internet Environment

4.1.1.2 This proposal

The sender's process determines the proper host for delivery given the recipient's unique ID. If the message is destined to the local network, delivery takes place as described earlier in this proposal. If the recipient is not local, the message may be passed to an MPM for foreign delivery. (A discussion of internet delivery which does not presuppose RFC 753 implementation is found later in this note.)

The environment in which the MPM operates does not assume any knowledge on the part of the local networks about addressees on foreign networks. Thus there are two possibilities which arise:

- The recipient has an ID known to the local networks.

  In this case, the local networks supply the RFC 753 "address". This can take place in the local networks' MPM or the user's sending or mail creation process.

- The recipient is unknown to the local networks.

  Here the sender must supply "mailbox" information himself, either explicitly or with help of his local host's database.

Thus, outgoing mail as described in this memo is compatible with RFC 753, with the benefit of reducing the burden on the MPM by handling mail deliveries that are local to local networks.

4.1.2 Messages in transit

Traffic between two MPMs is not affected by this proposal.

4.1.3 Incoming mail

The MPM on the networks local to the recipient will have access to the netmail database, allowing it to translate "mailboxes" to "addresses". It can determine the unique ID of the recipient (if not known), and initiate delivery to that recipient. Here RFC 753 and this proposal complement each other very well.

5. Implications of an internetwork message environment

The scheme described above is based upon the assumption that a unique identifier can be assigned to each registered recipient of mail. Whether or not this uniqueness can be guaranteed in a fairly unregulated internetwork environment is questionable. It is technically feasible, certainly. The difficulties are more political, because it is necessary to gain the cooperation of the administrators and user populations of foreign networks. Let's assume cooperation, however, and see what might happen in an internet environment.

5.1 Birthplaces

Each set of local networks would have its own database, for ease in access. It does not seem practical to register each ID in every database, however. That would be unnecessary, and would create access and storage problems at the network databases. Here the concept of a "birthplace", or ID origin, may be of use.

While an ID does not imply where the user is now, it can say something about who issued it. A simple system for determining the address for any ID can be maintained by having the issuing network keep a pointer for each ID it issues. One double indirection would yield the desired address, even if the ID were not issued on the local nets. A message originating on the local nets with an ID which is unknown to its database can be handled by determining the birthplace of the ID. An inquiry to the birthplace database would return a list of one or more networks on which the ID is registered. An inquiry to any of those would get the requisite information. All that is necessary to support this is for the birthplace record (small enough!) to be kept, and for the act of registration at a given net to automatically cause that net to notify the birthplace of the registration. (Conversely, a de-registration would cause a similar notification of the birthplace.)

5.1.1 ID resolution

The handling of ID resolution when the ID is not known to the local net does not seem to have a solution simpler than querying foreign nets until some success is achieved.

5.1.2 Hosts in an internet environment

The substitution of internet host names for simple host names should not cause any difficulty.

5.1.3 Orphans

Should a birthplace cease to exist (usually because its network is dismantled), it would be necessary for a second birthplace to "adopt" the first birthplace's records. Notification of this change could be propagated throughout the internet environment in much the same way as the addition of a new birthplace would be treated.

6. Conclusions

While ARPAnet message systems have been amazingly successful, there is much room for improvement in the quality and quantity of the services offered. Current protocols are limiting the development of new message systems. This paper has discussed a means of providing the underlying support necessary for building a new generation of message systems which can be better human-engineered in addition to providing more services and greater reliability.

Critics may argue that the proposal is too radical, too much of a departure from current practice. After all, today's message service is extremely straightforward in design, and therefore has comparatively few failure modes. The protocols in use have descended, with relatively few changes, from the first file transfer and message format protocols implemented on the ARPAnet. This makes them well understood; people are aware of both their shortcomings and usage. Finally, there are people who will not feel comfortable about requiring a network database, distrusting the reliability and questioning the possible cost of such a scheme.

On the other hand, it is undeniably true that very little more can be done to improve message services while staying within today's practices. New message systems which will be able to transmit facsimile, voice, and other media along with text require us to rethink message formats and do away with delivery protocols which are predicated upon the characteristics of ASCII text. The inception of internetwork message delivery causes us to re-evaluate how we handle messages locally. Finally, the USERNAME@HOST naming scheme has proved to be inadequate, while the divorce of recipients' identities from their locations seems a promising possibility as a replacement.

The ARPAnet will soon have a distributed database for supporting TIP Login. Only small, incremental costs would be associated with building and maintaining a netmail database at the same time. It can be argued that TIP Login requires at least the level of reliability required by a message delivery system. If the TIP Login database is successful, a netmail database can work, too.

It is clear that we will be implementing a new set of message format and delivery protocols in the near future, in order to allow for multi-media messages, internetwork message traffic, and the like. New message composition and delivery systems will be built to meet those specifications and take advantage of the avenues of development which they will open. If there will ever be an advantageous time to re-evaluate and re-design how messages are addressed and delivered, it is now, when we are about to enter upon an entirely new cycle of message composition and

delivery program implementation.

REFERENCES

[1] John F. Shoch.
Inter-Network Naming, Addressing, and Routing.
In Proceedings, COMPCON. IEEE Computer Society, Fall, 1979.

[2] Stephen Warshall.
On Names and Permissions.
Mass. Computer Associates. 1979.

[3] David H. Crocker, John J. Vittal, Kenneth T. Pogran,
D. Austin Henderson, Jr.
STANDARD FOR THE FORMAT OF ARPA NETWORK TEXT MESSAGES.
RFC 733, The Rand Corporation, Bolt Beranek and Newman Inc,
Massachussets Institute of Technology, Bolt Beranek and
Newman Inc., November, 1977.

[4] Jonathan B. Postel.
INTERNET MESSAGE PROTOCOL.
RFC 753, Information Sciences Institute, March, 1979.

[5] Robert H. Thomas, Paul J. Santos, and John F. Haverty.
TIP Login Notes.
Bolt Beranek and Newman. 1979.